

## ปฏิบัติการที่ 2 การปรับแก้โครงข่ายสามเหลี่ยมด้วยวิธีกำลังสองน้อยที่สุด

ปรับแก้ข้อมูลรังวัดโครงข่ายสามเหลี่ยมจากกล้อง Total Station ด้วยวิธีกฎเข็มทิศ (Compass Rule) และวิธีกำลังสองน้อยที่สุด (Least squares adjustment)

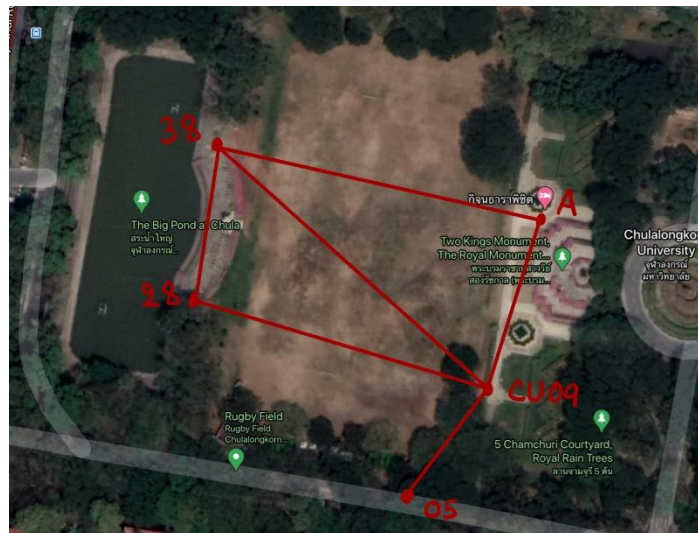
### วัตถุประสงค์

เปรียบเทียบผลลัพธ์การปรับแก้โครงข่ายสามเหลี่ยมด้วยวิธี Compass Rule กับ Least squares method

### วิธีปฏิบัติงาน

- 1) ลงพื้นที่เก็บข้อมูลภาคสนาม Traverse ด้วยกล้อง Total Station
- 2) นำข้อมูลมาทำการปรับแก้และหาค่าพิกัดหมุดที่ต้องการทราบพิกัด ด้วยวิธี Compass Rule
- 3) นำข้อมูลมาทำการปรับแก้และหาค่าพิกัดหมุดที่ต้องการทราบพิกัด ด้วยวิธี Least Square Adjustment ด้วยการเขียน Code ผ่าน โปรแกรม MatLab

### พื้นที่ปฏิบัติงาน และภาพสเก็ตโครงข่ายสามเหลี่ยม



รูปที่ 1 แสดงพื้นที่ปฏิบัติงานและโครงข่ายสามเหลี่ยม

ตารางที่ 1 สรุปผลการรังวัดวงรอบ (มุม)

มุม	AVG Angle (Degree)	STD Angle (Degree)
05 CU09 A	142.0417593	0.000290968
CU09 A 38	87.4572685	0.000229155
A 38 CU09	27.4122454	0.000360101
38 CU09 A	65.1259722	0.000283506
05 CU09 38	76.9136343	0.00028539
CU09 38 28	55.7819907	0.000130946
38 28 CU09	98.2091898	0.000267959
28 CU09 38	26.0082639	0.000259837

ตารางที่ 2 ผลการวัดระยะทาง

ระยะ	AVG Distance (m)	STD Distance (m)
05 - CU09	61.508	0.001258306
CU09 - A	58.438	0.00057735
A - 38	115.161	0.000816497
38 - CU09	126.805	0.000942809
38 - 28	56.184	0.000687184
28 - CU09	105.939	0.000687184

ผลการคำนวณ

การปรับแก้ด้วยวิธี Compass rule

ตารางที่ 3 ผลการปรับแก้มุมภายในวงรอบตามเงื่อนไข  $(n-2)*180^\circ$ 

1. สามเหลี่ยม CU09 – A – 38

Compass Rule				
Sta.	Hor Ang	Cor	Adj. Ang.	Azimuth
CU09				18.5112517
A	87.4572685	0.001504633	87.45877313	285.9700248
38	27.4122454	0.001504633	27.41375003	133.3837749
CU09	65.1259722	0.001504633	65.12747683	18.5112517
sum	179.9954861	0.0045139	180.0000000	
Closure	0.0045139			

## 2. สามเหลี่ยม CU09 – 38 – 28

Compass Rule				
Sta.	Hor Ang	Cor	Adj. Ang.	Azimuth
CU09				313.3831267
38	55.7819907	0.0001852	55.7821759	189.1653026
28	98.2091898	0.0001852	98.209375	107.3746776
CU09	26.0082639	0.0001852	26.0084491	313.3831267
sum	179.9994444	0.0005556	180.0000000	
Closure	0.0005556			

## ตารางที่ 4 ผลการปรับแก้วงรอบด้วยกฎเข็มทิศ

## 1. สามเหลี่ยม CU09 – A – 38

Sta.	Hor Dist	Azimuth	Latitude	Correct Lat	Departure	Correct Dep	Northing	Easting
CU09	58.438	18.5112517	55.4144953	0.000153224	18.5535321	0.00096245	1519282.160	665580.855
A	115.161	285.9700248	31.68475487	0.000301951	-110.71645	0.00189666	1519337.575	665599.409
38	126.805	133.3837749	-87.10003783	0.000332482	92.15797	0.00208843	1519369.260	665488.695
CU09							1519282.160	665580.855
sum	300.404		-0.000787657	0.000787657	-0.00494755	0.00494755		

คำนวณความถูกต้อง

$$\text{Linear miscloser} = 2.509865 \times 10^{-5} \text{ เมตร}$$

$$\text{Accuracy} = 1 : 11,968,932$$

## 2. สามเหลี่ยม CU09 – 38 – 28

Sta.	Hor Dist	Azimuth	Latitude	Correct Lat	Departure	Correct Dep	Northing	Easting
CU09	126.804	313.3831267	87.09830847	0.001661749	-92.1582285	0.0009425	1519282.160	665580.855
38	56.184	189.1653026	-55.46669356	0.000736284	-8.9491766	0.0004176	1519369.260	665488.698
28	105.939	107.3746776	-31.63540126	0.001388316	101.105258	0.00078742	1519313.794	665479.749
CU09							1519282.160	665580.855
sum	288.927		-0.003786349	0.003786349	-0.00214752	0.00214752		

คำนวณความถูกต้อง

$$\text{Linear miscloser} = 1.89483 \times 10^{-5} \text{ เมตร}$$

$$\text{Accuracy} = 1 : 15,248,187$$

ตารางที่ 5 ผลการปรับแก้ด้วยวิธีกำลังสองน้อยที่สุด

Point	Easting (m)	Northing (m)	STD Easting (m)	STD Northing (m)
38	665488.698	1519369.256	0.0005	0.0005
28	665479.749	1519313.791	0.0005	0.0005
A	665599.409	1519337.571	0.0006	0.0003

ตารางที่ 6 เปรียบเทียบค่าพิกัด

Sta.	Least Square		Compass Rule		Diff	
	Easting (m)	Northing (m)	Easting (m)	Northing (m)	dE	dN
38	665488.698	1519369.256	665488.695	1519369.260	0.003	0.004
28	665479.749	1519313.791	665479.749	1519313.794	0.000	0.003
A	665599.409	1519337.571	665599.410	1519337.575	0.001	0.004

สรุปผลการประมวลผลเปรียบเทียบค่าพิกัด

จากปฏิบัติการเปรียบเทียบผลการคำนวณปรับแก้ทั้งสองวิธีได้แก่ 1. กฎเข็มทิศ ( Compass Rule ) และ 2. วิธีกำลังสองน้อยที่สุด ( Least Square ) พบว่าทั้งสองวิธีสามารถปรับแก้วงรอบให้มีความถูกต้องตามทฤษฎีทางเรขาคณิต และได้ผลที่ใกล้เคียงกัน โดย Compass Rule เป็นวิธีที่ง่ายและไม่จำเป็นต้องใช้ความรู้การปรับแก้มากนัก แต่สำหรับ Least Square จำเป็นต้องใช้ความรู้ Matrix , การเขียนโปรแกรม , และความเข้าใจเรื่อง Azimuth เพื่อนำมาสร้าง Observation Equation แต่ในทางทฤษฎี Least Square เป็นวิธีที่มีประสิทธิภาพและความน่าเชื่อถือมากกว่า เหมาะกับงานที่ต้องการความน่าเชื่อถือสูง หรืองานรังวัดที่มีข้อมูลจำนวนมาก

## Matlab Code

```

%Least Square Adjustment for Traverse with Triangle Network
%Data Corrected by Party4
%This code is work for Loop1
clc
syms E38 N38 E28 N28 Ea Na
Ecu = 665580.855; %Known coordinate point
Ncu = 1519282.156;

%Observation Equation
%atan must be careful about angle
%tan( <90 degree )
%Then draw the map before made an equation.
% Azimuth CU09-05 is 313.3831267 Degree.
v1 = sqrt((Ecu-E38)^2 + (Ncu-N38)^2)-126.804;
v2 = sqrt((E38-E28)^2 + (N38-N28)^2)-56.184;
v3 = sqrt((Ecu-E28)^2 + (Ncu-N28)^2)-105.939;
v4 = atan((N38-Ncu) / (Ecu-E38)) - (313.3831267-270)*pi/180.0;
v5 = atan((N28-Ncu) / (Ecu-E28)) - (107.3746776-90)*pi/180.0;
v6 = atan((E38-E28) / (N38-N28)) - (189.1653026-180)*pi/180.0;

fL = [v1;v2;v3;v4;v5;v6];
%Start value for put in adjustment loop
%These value from compass rule adjustment , It's quite accurate.
iE38 = 665488.6977 ;
iN38 = 1519369.26 ;
iE28 = 665479.749 ;
iN28 = 1519313.794 ;

mA = jacobian(fL,[E38 N38 E28 N28]);
%Weight matrix made from SD from surveying 3 sets
%first three are Distance SD
%last three are Angle SD
W = diag([1.0/(0.000745355992503489)^2 1.0/(0.000687184270937547)^2 1.0/(0.000687184270939558)^2
1/(0.00028539*pi/180)^2 1/(0.000267959*pi/180)^2 1/(0.000130946*pi/180)^2]);

for i = 1:8
    A = double(subs(mA,[E38 N38 E28 N28],[iE38, iN38, iE28, iN28]));
    L = double(subs(fL,[E38 N38 E28 N28],[iE38, iN38, iE28, iN28]));
    %subs = replace parameter by value
    N = A'*W*A;
    U = A'*W*L;
    dx = -1*N\U;
    iE38 = iE38 + dx(1);
    iN38 = iN38 + dx(2);
    iE28 = iE28 + dx(3);
    iN28 = iN28 + dx(4);
    % This line for checking convergent or not
    fprintf("E38=%.3f, N38=%.3f E28=%.3f N28=%.3f\n",dx(1),dx(2),dx(3),dx(4))
end
fprintf("Coordinate in WGS84/UTM47N (m) \n")
fprintf("E38 = %.3f, N38 = %.3f E28 = %.3f N28 = %.3f\n",iE38,iN38,iE28,iN28);
var_cov = diag(inv(N));
std = sqrt(var_cov);
fprintf("STD-E38 = %.4f, STD_N38 = %.4f STD_E28 = %.4f STD_N28 = %.4f\n",std(1),std(1),std(2),std(3));

```

## Result

Coordinate in WGS84/UTM47N (m)

E38 = 665488.698, N38 = 1519369.256 E28 = 665479.749 N28 = 1519313.791

STD-E38 = 0.0005, STD\_N38 = 0.0005 STD\_E28 = 0.0005 STD\_N28 = 0.0005

## Matlab Code

```
%Least Square Adjustment for Traverse with Triangle Network
%Data Corrected by Party3
%This code is work for loop2
clc
syms E38 N38 E28 N28 Ea Na
Ecu = 665580.855;
Ncu = 1519282.156;

%Observation Equation
%atan must be careful about angle
%tan( <90 degree )
%Then draw the map before made an equation.
% Azimuth CU09-05 is 313.3831267 Degree.
v1 = sqrt((Ecu-E38)^2 + (Ncu-N38)^2)-126.804;
v2 = sqrt((Ea-Ecu)^2 + (Na-Ncu)^2)-58.438;
v3 = sqrt((Ea-E38)^2 + (Na-N38)^2)-115.161;
v4 = atan((Ea-Ecu) / (Na-Ncu)) - 18.5112517*pi/180.0 ;
v5 =atan((N38-Na) / (Ea-E38)) -(285.9700248-270)*pi/180.0 ;
v6 =atan((N38-Ncu) / (Ecu-E38)) - (133.3837749 - 90)*pi/180.0 ;

fL = [v1;v2;v3;v4;v5;v6];
%Start value for put in adjustment loop
%These value from compass rule adjustment , It's quite accurate.
iE38 = 665488.6977 ;
iN38 = 1519369.26 ;
iE28 = 665479.749 ;
iN28 = 1519313.794 ;
iEa = 665599.409;
iNa = 1519337.57 ;
u = [E38 N38 Ea Na];
mA = jacobian(fL,u);
%Weight matrix made from SD from surveying 3 sets
%first three are Distance SD
%last three are Angle SD
W = diag([1.0/(0.000942809)^2 1.0/(0.00057735)^2 1.0/(0.000816497)^2 1/(0.000290968*pi/180)^2
1/(0.000229155*pi/180)^2 1/(0.000360101*pi/180)^2]);

for i = 1:8
    A = double(subs(mA,u,[iE38, iN38, iEa, iNa]));
    L = double(subs(fL,u,[iE38, iN38, iEa, iNa]));
    %subs = replace parameter by value
    N = A'*W*A;
    U = A'*W*L;
    dx = -1*N\U;
    iE38 = iE38 + dx(1);
    iN38 = iN38 + dx(2);
    iEa = iEa + dx(3);
    iNa = iNa + dx(4);
    % This line for checking convergent or not
    %fprintf("E38=%.3f, N38=%.3f Ea=%.3f Na=%.3f\n",dx(1),dx(2),dx(3),dx(4))
end
fprintf("Coordinate in WGS84/UTM47N (m) \n")
fprintf("E38 = %.3f, N38 = %.3f Ea = %.3f Na = %.3f\n",iE38,iN38,iEa,iNa);
var_cov = diag(inv(N));
std = sqrt(var_cov);
fprintf("STD-E38 = %.4f, STD_N38 = %.4f STD_Ea = %.4f STD_Na = %.4f\n",std(1),std(1),std(2),std(3));
```

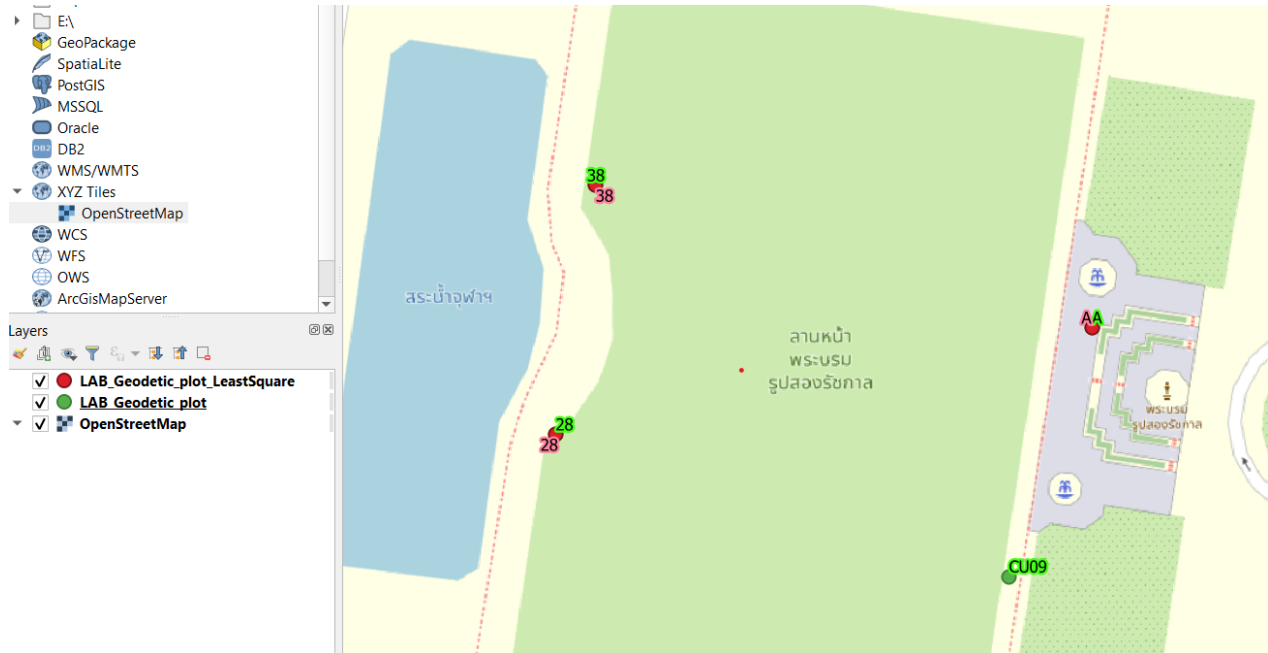
## Result

Coordinate in WGS84/UTM47N (m)

E38 = 665488.695, N38 = 1519369.255 Ea = 665599.409 Na = 1519337.571

STD-E38 = 0.0006, STD\_N38 = 0.0006 STD\_Ea = 0.0006 STD\_Na = 0.0003

เมื่อนำข้อมูลพิกัดการปรับแก้ทั้งสอง มาแสดงใน QGIS  
จะเห็นว่าพิกัดจากการปรับแก้ทั้งสองอยู่ใกล้กันมาก



ข้อมูลอื่น ไฟล์ Excel และ MatLab code ดูเพิ่มเติมได้จาก

<https://github.com/TonAsianMaster/Geodetic-Survey-Traverse-Adj>

